

Common Sensor Data Record Science Processing Algorithm (C-SDR_SPA) User's Guide

Version 2.2

April 2017



**GODDARD SPACE FLIGHT CENTER
GREENBELT, MARYLAND**

Common Sensor Data Record Science Processing Algorithm

C-SDR_SPA

General

The NASA Goddard Space Flight Center's (GSFC) Direct Readout Laboratory (DRL), Code 606.3 developed this software for the International Polar Orbiter Processing Package (IPOPP). IPOPP maximizes the utility of Earth science data for making real-time decisions by giving fast access to instrument data and derivative products from the Suomi National Polar-orbiting Partnership (SNPP), Aqua, and Terra missions and, in the future, the Joint Polar Satellite System (JPSS) mission.

Users must agree to all terms and conditions in the Software Usage Agreement on the DRL Web Portal before downloading this software.

Software and documentation published on the DRL Web Portal may occasionally be updated or modified. The most current versions of DRL software are available at the DRL Web Portal:

<http://directreadout.sci.gsfc.nasa.gov/?id=software>

Questions relating to the contents or status of this software and its documentation should be addressed to the DRL via the Contact DRL mechanism at the DRL Web Portal:

<http://directreadout.sci.gsfc.nasa.gov/?id=dspContent&cid=66>

Algorithm Wrapper Concept

The DRL has developed an algorithm wrapper to provide a common command and execution interface to encapsulate multi-discipline, multi-mission science processing algorithms. The wrapper also provides a structured, standardized technique for packaging new or updated algorithms with minimal effort.

A Science Processing Algorithm (SPA) is defined as a wrapper and its contained algorithm. SPAs will function in a standalone, cross-platform environment to serve the needs of the broad Direct Readout community. Detailed information about SPAs and other DRL technologies is available at the DRL Web Portal.

Software Description

This software package contains the Common Sensor Data Record Science Processing Algorithm (C-SDR_SPA). The C-SDR_SPA software package processes Suomi NPP Visible Infrared Imaging Radiometer Suite (VIIRS), Advanced Technology Microwave Sounder (ATMS), and Cross-track Infrared Sounder (CrIS) Raw Data Record (RDR) HDF5 products into corresponding instrument-specific and mission-compliant HDF5 Sensor Data Record (SDR) and Geolocation swath products. The SPA functions in two modes: standalone, or as an IPOPP plug-in.

Software Version

Version 2.2 of the DRL algorithm wrapper was used to package the SPA described in this document. The C-SDR_SPA package contains the Interface Data Processing Segment (IDPS) Direct Readout (DRO) Software. The IDPS DRO Software leverages the following versions of the IDPS Operation System (OPS) Software:

- a) VIIRS SDR, 1.5.08.10;
- b) CrIS SDR, 1.5.08.10;
- c) ATMS SDR, 1.5.08.10.

The DRO Algorithm software is a standalone process consisting of reuse from Algorithm Development Library (ADL), Processing Subsystem (PRO), Data Delivery Subsystem (DDS), and Ingest Subsystem (ING) code, which has been extended and specialized with additional code. This software, collectively referred to as DRO, is an implementation of NPP Algorithm support using NPP ADL packaging for standalone use in a Direct Readout environment.

Enhancements to this SPA include:

- update to v1.5.08.10 (Mx8.10) of the official operational SDR software;
- CrIS SDR extended interferogram capability (provides further flexibility in the HRD downlink configuration);
- update to the VIIRS SDR algorithm for enhanced processing of data with severe loss and degradation;
- updated ATMS, CrIS, and VIIRS Look-Up Tables (LUTs).

This software will execute on a 64-bit computer and has been tested on computers with 32GB of RAM, with the following operating systems:

- a) Fedora 20 X86_64;
- b) CentOS Linux 7 X86_64;
- c) OpenSUSE Linux 13.2 X86_64;
- d) Kubuntu 14.04 X86_64.

Copyright 1999-2007, United States Government as represented by the Administrator for the National Aeronautics and Space Administration. All Rights Reserved.

Credits

The Direct Readout SDR algorithms within C-SDR_SPA were provided to the DRL by the JPSS Ground Project.

Prerequisites

To run this package, you must have the Java Development Kit (JDK) or Java Runtime Engine (JRE) (Java 1.6.0_25 or higher) installed on your computer, and have the Java installation bin/ subdirectory in your PATH environment variable. You must also have Perl installed on

your computer (5.12.4 or higher; also install the Perl Data Dumper if it is not included in your computer's Perl installation). This package contains 64-bit binaries statically pre-compiled on an x86-compatible 64-bit computer running under CentOS 7, using gcc4.5.1. The C-SDR_SPA also requires at least 8 GB of memory to run successfully, although more is recommended for improved performance.

Program Inputs and Outputs

This SPA includes SDR algorithms for the SNPP VIIRS, CrIS, and ATMS instruments.

The VIIRS SDR algorithm takes a VIIRS RDR file (containing a VIIRS Science RDR and a Spacecraft Diary RDR) and required ancillaries as input and outputs the VIIRS Imagery resolution SDRs, VIIRS Moderate resolution SDRs, the VIIRS Day/Night Band (DNB) SDR, the VIIRS On Board Calibrator Intermediate Product (IP), the VIIRS Calibrated Dual Gain IP, and the VIIRS Geolocation products.

The ATMS SDR algorithm takes an ATMS RDR file (containing an ATMS Science RDR and a Spacecraft Diary RDR) and required ancillaries as input and outputs the ATMS SDR, the ATMS Temperature Data Record (TDR), and the ATMS Geolocation products.

The CrIS SDR algorithm takes a CrIS RDR file (containing a CrIS Science RDR and a Spacecraft Diary RDR) and required ancillaries as input and outputs the CrIS SDR and Geolocation products.

Installation and Configuration

Installing as a Standalone Application:

Download the C-SDR_2.2_SPA_2.2.tar.gz and C-SDR_2.2_SPA_2.2_testdata.tar.gz (optional) files into the same directory.

Decompress and un-archive the C-SDR_2.2_SPA_2.2.tar.gz and C-SDR_2.2_SPA_2.2_testdata.tar.gz (optional) files:

```
$ tar -xzf C-SDR_2.2_SPA_2.2.tar.gz  
$ tar -xzf C-SDR_2.2_SPA_2.2_testdata.tar.gz
```

This will create the following subdirectories:

SPA
 C-SDR
 algorithm
 ancillary
 mode
 station
 testdata
 testscripts
 wrapper

Installing into an IPOPP Framework: This SPA can also be installed dynamically into an IPOPP framework to automate production of ATMS SDR/TDR/Geolocation, CrIS SDR/Geolocation, and VIIRS SDR/Geolocation products. The SPA installation process will install SPA service(s) into IPOPP. An SPA service is an IPOPP agent that provides the mechanism necessary for running an SPA automatically within the IPOPP framework. Once this SPA is installed, users must enable the service(s) corresponding to this SPA along with any other prerequisite service(s). Instructions for installing an SPA and enabling its services are contained in the IPOPP User's Guide (available on the DRL Web Portal). The SPA services associated with this SPA are listed in Appendix A.

Software Package Testing and Validation

The testscripts subdirectory contains test scripts that can be used to verify that your current installation of the SPA is working properly, as described below. Note that the optional C-SDR_2.2_SPA_2.2testdata.tar.gz file is required to execute these testing procedures.

Step 1: cd into the testscripts directory.

Step 2: There are scripts named run-atms.sh, run-cris.sh, and run-viirs.sh inside the testscripts directory.

To run the VIIRS SDR algorithm, use

```
$ ./run-viirs.sh
```

A successful execution usually requires 2 minutes or more, depending on the speed of your computer and the size of the input. If everything is working properly, the script will terminate with a message such as:

```
Output viirs.gdnbo is /home/ipopp/SPA/C-SDR/testdata/output/GDNBO.h5
Output viirs.gimgo is /home/ipopp/SPA/C-SDR/testdata/output/GIMGO.h5
Output viirs.gitco is /home/ipopp/SPA/C-SDR/testdata/output/GITCO.h5
Output viirs.gmodo is /home/ipopp/SPA/C-SDR/testdata/output/GMODO.h5
Output viirs.gmtco is /home/ipopp/SPA/C-SDR/testdata/output/GMTCO.h5
Output viirs.icdbg is /home/ipopp/SPA/C-SDR/testdata/output/ICDBG.h5
Output viirs.ivcdb is /home/ipopp/SPA/C-SDR/testdata/output/IVCDB.h5
Output viirs.ivobc is /home/ipopp/SPA/C-SDR/testdata/output/IVOBC.h5
Output viirs.svndb is /home/ipopp/SPA/C-SDR/testdata/output/SVNDB.h5
Output viirs.svi01 is /home/ipopp/SPA/C-SDR/testdata/output/SVI01.h5
Output viirs.svi02 is /home/ipopp/SPA/C-SDR/testdata/output/SVI02.h5
Output viirs.svi03 is /home/ipopp/SPA/C-SDR/testdata/output/SVI03.h5
Output viirs.svi04 is /home/ipopp/SPA/C-SDR/testdata/output/SVI04.h5
Output viirs.svi05 is /home/ipopp/SPA/C-SDR/testdata/output/SVI05.h5
Output viirs.svm01 is /home/ipopp/SPA/C-SDR/testdata/output/SVM01.h5
Output viirs.svm02 is /home/ipopp/SPA/C-SDR/testdata/output/SVM02.h5
Output viirs.svm03 is /home/ipopp/SPA/C-SDR/testdata/output/SVM03.h5
Output viirs.svm04 is /home/ipopp/SPA/C-SDR/testdata/output/SVM04.h5
Output viirs.svm05 is /home/ipopp/SPA/C-SDR/testdata/output/SVM05.h5
Output viirs.svm06 is /home/ipopp/SPA/C-SDR/testdata/output/SVM06.h5
Output viirs.svm07 is /home/ipopp/SPA/C-SDR/testdata/output/SVM07.h5
Output viirs.svm08 is /home/ipopp/SPA/C-SDR/testdata/output/SVM08.h5
Output viirs.svm09 is /home/ipopp/SPA/C-SDR/testdata/output/SVM09.h5
```

```
Output viirs.svm10 is /home/ipopp/SPA/C-SDR/testdata/output/SVM10.h5
Output viirs.svm11 is /home/ipopp/SPA/C-SDR/testdata/output/SVM11.h5
Output viirs.svm12 is /home/ipopp/SPA/C-SDR/testdata/output/SVM12.h5
Output viirs.svm13 is /home/ipopp/SPA/C-SDR/testdata/output/SVM13.h5
Output viirs.svm14 is /home/ipopp/SPA/C-SDR/testdata/output/SVM14.h5
Output viirs.svm15 is /home/ipopp/SPA/C-SDR/testdata/output/SVM15.h5
Output viirs.svm16 is /home/ipopp/SPA/C-SDR/testdata/output/SVM16.h5
```

To run the ATMS SDR algorithm, use

```
$ ./run-atms.sh
```

A successful execution usually requires 1 minute or more, depending on the speed of your computer and the size of the input. If everything is working properly, the script will terminate with a message such as:

```
Output atms.fatms is /home/ipopp/SPA/C-SDR/testdata/output/FATMS.h5
Output atms.gatmo is /home/ipopp/SPA/C-SDR/testdata/output/GATMO.h5
Output atms.satms is /home/ipopp/SPA/C-SDR/testdata/output/SATMS.h5
Output atms.tatms is /home/ipopp/SPA/C-SDR/testdata/output/TATMS.h5
```

To run the CrIS SDR algorithm, use

```
$ ./run-cris.sh
```

A successful execution usually requires 7 minutes or more, depending on the speed of your computer and the size of the input. If everything is working properly, the script will terminate with a message such as:

```
Output cris.gcrso is /home/ipopp/SPA/C-SDR/testdata/output/GCRSO.h5
Output cris.rgcrs is /home/ipopp/SPA/C-SDR/testdata/output/RGCRS.h5
Output cris.rgtrs is /home/ipopp/SPA/C-SDR/testdata/output/RGTRS.h5
Output cris.scris is /home/ipopp/SPA/C-SDR/testdata/output/SCRIS.h5
```

You can cd to the output directory to verify that the science products exist. Test output product(s) are available for comparison in the testdata/output directory. These test output product(s) were generated on a 64-bit PC architecture computer running CentOS 7. The output products serve as an indicator of expected program output. Use a comparison utility (such as diff, h5diff, etc.) to compare your output product(s) to those provided in the testdata/output directory. Locally generated files may differ slightly from the provided output files because of differences in machine architecture or operating systems.

If there is a problem and the code terminates abnormally, the problem can be identified using the log files. Log files are automatically generated within the directory used for execution. They start with stdfile* and errfile*. Other log and intermediate files may be generated automatically within the directory used for execution. They are useful for traceability and debugging purposes. However it is strongly recommended that users clean up log files and intermediate files left behind in the run directory before initiating a fresh execution of the SPA. Intermediate files from a previous run may affect a successive run and produce ambiguous

results. Please report any errors that cannot be fixed to the DRL.

Program Operation

In order to run the package using your own input data, you can either use the run scripts within the wrapper subdirectories, or modify the test scripts within the testscripts subdirectory.

To Use the Run Scripts

Identify the 'run' scripts: The wrapper directory within this package contains subdirectories named ATMS_C-SDR, CRIS_C-SDR, and VIIRS_C-SDR. The subdirectories contain an executable called 'run'. Execute 'run' within the correct wrapper subdirectory to generate the corresponding product. For instance, the 'run' within wrapper/VIIRS_C-SDR is used for creating VIIRS SDR and Geolocation outputs. Note that to execute 'run', you need to have java on your path.

Specify input parameters using <label value> pairs: To execute the 'run' scripts, you must supply the required input and output parameters. Input and output parameters are usually file paths or other values (e.g., an automatic search flag). Each parameter is specified on the command line by a <label value> pair. Labels are simply predefined names for parameters. Each label must be followed by its actual value. Each process has its own set of <label value> pairs that must be specified in order for it to execute. Some of these pairs are optional, meaning the process would still be able to execute even if that parameter is not supplied. The types of <label value> pairs that the C-SDR_SPA uses are:

- a) Input file label/values. These are input file paths. Values are absolute or relative paths to the corresponding input file.
- b) Output file label/values. These are output files that are produced by the SPA. Values are absolute or relative paths of the files you want to generate.
- c) Optional parameter label/values. These are parameters that may be optionally passed to the SPA (e.g., enabling/disabling compression of output products).

This version of the C-SDR_SPA provides an optional interface for cross-granule RDR file inputs to the VIIRS SDR algorithm. Use of cross-granule RDR file inputs improves processing at the granule boundaries of the dual-gain (M1 – M5, M7, M13) SDR output products. The granule start/end boundaries of the optional cross-granule RDR file inputs must be temporally adjacent (not overlapping) to the granule start/end boundaries of the primary RDR file input. The "previous" RDR file should be the one that is adjacent and precedes the first granule of the primary RDR file while the "next" RDR file should be the one that is adjacent and follows the last granule of the primary RDR file input.

To provide the optional cross-granule RDR file inputs, the files need to be specified as input parameters by using the optional ".prev" and ".next" file labels; the file label designated as ".prev" corresponds to the cross-granule RDR file that is temporally adjacent and precedes the first granule of the primary RDR file, and ".next" corresponds to the cross-granule RDR file that is temporally adjacent and follows the last granule of the primary RDR file.

The following tables contain labels, and their descriptions, required by the C-SDR_SPA.

VIIRS_C-SDR

Input File Labels	Description	Source
viirs.rdr Optional: viirs.rdr.prev viirs.rdr.next	VIIRS Raw Data Record (RDR) file	<p>1. Real time VIIRS RDR products over the eastern US region are available from the DRL ftp site at: ftp://is.sci.gsfc.nasa.gov/gsfcdata/npp/viirs/level0/RNSCA-RVIRS_npp_dyyyymmdd_thhmmssS_ehhmmssS*.h5</p> <p>Where yyyy, mm, dd represents the year, month, and day of month for the start of the swath; the first hh, mm, ss, S represents the hour, minutes, seconds, and 10th of a second for the start of the swath and the second hh, mm, ss, S represents the end time of the swath.</p> <p>2. VIIRS RDR products for other locations and times are available for download at: www.class.noaa.gov</p>
tle	Two Line Element file	<p>For recent TLE files go to: ftp://is.sci.gsfc.nasa.gov/ancillary/ephemeris/tle/drl.tle.yyyyymmddhh</p> <p>For archived TLE files go to: ftp://is.sci.gsfc.nasa.gov/ArchivedAncillary/ephe meris/tle/drl.tle.yyyyymmddhh</p> <p>Where yyyy, mm, dd, hh represents the year, month, day of month and hour for the tle ancillary file.</p>
polar	Polar Wander ancillary file	<p>DRL ftp site for Polar Wander files: ftp://is.sci.gsfc.nasa.gov/ancillary/temporal/off_USNO-PolarWander-UT1-ANC_Ser7_USNO_000f_yyyyymmdd*.ascii</p> <p>Archived Polar Wander files: ftp://is.sci.gsfc.nasa.gov/ArchivedAncillary/temp oral/off_USNO-PolarWander-UT1-ANC_Ser7_USNO_000f_yyyyymmdd*.ascii</p> <p>Where yyyy, mm, dd represents the year, month, and day for the polar wander ancillary file.</p>
sdr.lut (optional)	VIIRS Look Up Tables (LUTs) compressed tar file	DRL ftp site for VIIRS LUTs: ftp://is.sci.gsfc.nasa.gov/ancillary/LUTs/npp/viirs/1.5.07.01/VIIRS-SDR_BE_LUTs_yyyyymmdd.tar.gz

Input File Labels	Description	Source
		<p>Archived VIIRS LUTs: ftp://is.sci.gsfc.nasa.gov/ArchivedAncillary/LUTs/npp/viirs/VIIRS-SDR_BE_LUTs_yymmdd.tar.gz</p> <p>Where yy, mm, dd represents the year, month, and day for the LUT collection file.</p>
leapsec	Leapsec ancillary file	<p>DRL ftp site for leapsec files: ftp://is.sci.gsfc.nasa.gov/ancillary/temporal/leapsec.yyyymmddhh.dat</p> <p>Where yyyy, mm, dd, hh represents the year, month, day, and hour for the leapsec ancillary file.</p>

Optional Parameters	Description
compress	<p>Parameter to enable or disable compression of C-SDR_SPA's HDF5 output products. Can be (a) 'on' to enable compression (gzip level 5 compression w/ shuffling algorithm applied) or (b) 'off' to disable compression.</p> <p>If this optional parameter is not provided, C-SDR_SPA uses 'off' by default and produces non-compressed HDF5 output products.</p>

Output File Labels	Description
viirs.gimgo	VIIRS I-Band Geolocation output HDF file path
viirs.gitco	VIIRS I-Band Geolocation terrain-corrected output HDF file path
viirs.gmodo	VIIRS M-Band Geolocation output HDF file path
viirs.gmtco	VIIRS M-Band Geolocation terrain-corrected output HDF file path
viirs.gdnbo	VIIRS Day/Night Band Geolocation output HDF file path
viirs.icdbg	VIIRS M-Band Unaggregated Geolocation output HDF file path
viirs.svdbn	VIIRS Day/Night Band SDR output HDF file path
viirs.svixx {xx= 01 to 05}	VIIRS 375m Ix {x=01 to 05} Band SDR output HDF file path
viirs.svmxx {xx= 01 to 16}	VIIRS 750m Mx {x=01 to 16} Band SDR output HDF file path
viirs.ivobc	VIIRS On-board Calibrator IP output HDF file path
viirs.ivcdb	VIIRS Calibrator Dual-Gain IP output HDF file path

ATMS_C-SDR

Input File Labels	Description	Source
atms.rdr	ATMS Raw Data Record (RDR) file	<ol style="list-style-type: none"> Real time ATMS RDR products over the eastern US region are available from the DRL ftp site at: ftp://is.sci.gsfc.nasa.gov/gsfcdata/npp/atms/level0/RATMS-RNSCA_npp_dyyyymmdd_thhmmssS_ehhmmssS*.h5 Where yyyy, mm, dd represents the year, month, and day of month for the start of the swath; the first hh, mm, ss, S represents the hour, minutes, seconds, and 10th of a second for the start of the swath and the second hh, mm, ss, S represents the end time of the swath. ATMS RDR products for other locations and times are available for download at: www.class.noaa.gov
tle	Two Line Element file	For recent TLE files go to: ftp://is.sci.gsfc.nasa.gov/ancillary/ephemeris/tle/

Input File Labels	Description	Source
		drl.tle.yyyymmddhh For archived TLE files go to: ftp://is.sci.gsfc.nasa.gov/ArchivedAncillary/ephemeris/tle/drl.tle.yyyymmddhh Where yyyy, mm, dd, hh represents the year, month, day of month and hour for the tle ancillary file.
polar	Polar Wander ancillary file	DRL ftp site for Polar Wander files: ftp://is.sci.gsfc.nasa.gov/ancillary/temporal/off_USNO-PolarWander-UT1-ANC_Ser7_USNO_000f_yyyymmdd*.ascii Archived Polar Wander files: ftp://is.sci.gsfc.nasa.gov/ArchivedAncillary/temporal/off_USNO-PolarWander-UT1-ANC_Ser7_USNO_000f_yyyymmdd*.ascii Where yyyy, mm, dd represents the year, month, and day for the polar wander ancillary file.
sdr.lut (optional)	ATMS Look Up Tables (LUTs) compressed tar file	DRL ftp site for ATMS LUTs: ftp://is.sci.gsfc.nasa.gov/ancillary/LUTs/npp/atms/ATMS-SDR_BE_LUTs_yymmdd.tar.gz Archived ATMS LUTs: ftp://is.sci.gsfc.nasa.gov/ArchivedAncillary/LUTs/npp/atms/ATMS-SDR_BE_LUTs_yymmdd.tar.gz Where yy, mm, dd represents the year, month, and day for the LUT collection file.
leapsec	Leapsec ancillary file	DRL ftp site for leapsec files: ftp://is.sci.gsfc.nasa.gov/ancillary/temporal/leapsec.yyyymmddhh.dat Where yyyy, mm, dd, hh represents the year, month, day, and hour for the leapsec ancillary file.

Optional Parameters	Description
compress	Parameter to enable or disable compression of C-SDR_SPA's HDF5 output products. Can be (a) 'on' to enable compression (gzip level 5 compression w/ shuffling algorithm applied) or (b) 'off' to disable compression. If this optional parameter is not provided, C-SDR_SPA uses 'off' by default and produces non-compressed HDF5 output products.

Output File Labels	Description
atms.satms	ATMS SDR output HDF file path
atms.fatms	ATMS SDR (full-sized, floating-point version) output HDF file path
atms.tatms	ATMS TDR output HDF file path
atms.gatmo	ATMS Geolocation output HDF file path

CRIS_C-SDR

Input File Labels	Description	Source
cris.rdr	CrIS Raw Data Record (RDR) file	<p>1. Real time CrIS RDR products over the eastern US region are available from the DRL ftp site at: ftp://is.sci.gsfc.nasa.gov/gsfcdata/npp/cris/level0/RCRIS-RNSCA_npp_dyyyymmdd_thhmmssS_ehhmmssS*.h5</p> <p>Where yyyy, mm, dd represents the year, month, and day of month for the start of the swath; the first hh, mm, ss, S represents the hour, minutes, seconds, and 10th of a second for the start of the swath and the second hh, mm, ss, S represents the end time of the swath.</p> <p>2. CrIS RDR products for other locations and times are available for download at: www.class.noaa.gov</p>
tle	Two Line Element file	<p>For recent TLE files go to: ftp://is.sci.gsfc.nasa.gov/ancillary/ephemeris/tle/drl.tle.yyyyymmddhh</p> <p>For archived TLE files go to: ftp://is.sci.gsfc.nasa.gov/ArchivedAncillary/ephemeris/tle/drl.tle.yyyyymmddhh</p> <p>Where yyyy, mm, dd, hh represents the year, month, day of month and hour for the tle ancillary file.</p>
polar	Polar Wander ancillary file	<p>DRL ftp site for Polar Wander files: ftp://is.sci.gsfc.nasa.gov/ancillary/temporal/off_USNO-PolarWander-UT1-ANC_Ser7_USNO_000f_yyyyymmdd*.ascii</p> <p>Archived Polar Wander files: ftp://is.sci.gsfc.nasa.gov/ArchivedAncillary/temporal/off_USNO-PolarWander-UT1-ANC_Ser7_USNO_000f_yyyyymmdd*.ascii</p> <p>Where yyyy, mm, dd represents the year, month, and day for the polar wander ancillary file.</p>
sdr.lut (optional)	CrIS Look Up Tables (LUTs) compressed tar file	<p>DRL ftp site for CrIS LUTs: ftp://is.sci.gsfc.nasa.gov/ancillary/LUTs/npp/cris/CRIS-SDR_BE_LUTs_yymmdd.tar.gz</p> <p>Archived CrIS LUTs:</p>

Input File Labels	Description	Source
		<p>ftp://is.sci.gsfc.nasa.gov/ArchivedAncillary/LUTs/npp/cris/CRIS-SDR_BE_LUTs_yymmdd.tar.gz</p> <p>Where yy, mm, dd represents the year, month, and day for the LUT collection file.</p>
leapsec	Leapsec ancillary file	<p>DRL ftp site for leapsec files: ftp://is.sci.gsfc.nasa.gov/ancillary/temporal/leapsec.yyyymmddhh.dat</p> <p>Where yyyy, mm, dd, hh represents the year, month, day, and hour for the leapsec ancillary file.</p>

Optional Parameters	Description
compress	<p>Parameter to enable or disable compression of C-SDR_SPA's HDF5 output products. Can be (a) 'on' to enable compression (gzip level 5 compression w/ shuffling algorithm applied) or (b) 'off' to disable compression.</p> <p>If this optional parameter is not provided, C-SDR_SPA uses 'off' by default and produces non-compressed HDF5 output products.</p>

Output File Labels	Description
cris.scris	CrIS SDR output HDF file path
cris.gcrso	CrIS Geolocation output HDF file path
cris.rgcrs	CrIS Geolocation (radians version) output HDF file path
cris.rgtrs	CrIS Geolocation (radians and terrain-corrected version) output HDF file path

Execute the 'run': The following script shows an example of a command line to run the VIIRS SDR algorithm from the testscripts directory:

```
$ ./wrapper/VIIRS_C-SDR/run \
viirs.rdr \
./testdata/input/RNSCA-RVIRS_npp_d20150828_t1953514_e1955167_b19869_c20150831164852597445_noaa_ops.h5 \
viirs.gdnbo ./testdata/output/GDNBO.h5 \
viirs.gimgo ./testdata/output/GIMGO.h5 \
viirs.gitco ./testdata/output/GITCO.h5 \
viirs.gmodo ./testdata/output/GMODO.h5 \
viirs.gmtco ./testdata/output/GMTCO.h5 \
viirs.icdbg ./testdata/output/ICDBG.h5 \
viirs.ivcdb ./testdata/output/IVCDB.h5 \
viirs.ivobc ./testdata/output/IVOBC.h5 \
viirs.svddb ./testdata/output/SVDDNB.h5 \
viirs.svi01 ./testdata/output/SVI01.h5 \
viirs.svi02 ./testdata/output/SVI02.h5 \
```

```

viirs.svi03 ..../testdata/output/SVI03.h5 \
viirs.svi04 ..../testdata/output/SVI04.h5 \
viirs.svi05 ..../testdata/output/SVI05.h5 \
viirs.svm01 ..../testdata/output/SVM01.h5 \
viirs.svm02 ..../testdata/output/SVM02.h5 \
viirs.svm03 ..../testdata/output/SVM03.h5 \
viirs.svm04 ..../testdata/output/SVM04.h5 \
viirs.svm05 ..../testdata/output/SVM05.h5 \
viirs.svm06 ..../testdata/output/SVM06.h5 \
viirs.svm07 ..../testdata/output/SVM07.h5 \
viirs.svm08 ..../testdata/output/SVM08.h5 \
viirs.svm09 ..../testdata/output/SVM09.h5 \
viirs.svm10 ..../testdata/output/SVM10.h5 \
viirs.svm11 ..../testdata/output/SVM11.h5 \
viirs.svm12 ..../testdata/output/SVM12.h5 \
viirs.svm13 ..../testdata/output/SVM13.h5 \
viirs.svm14 ..../testdata/output/SVM14.h5 \
viirs.svm15 ..../testdata/output/SVM15.h5 \
viirs.svm16 ..../testdata/output/SVM16.h5 \
sdr.lut ..../testdata/input/VIIRS-SDR_BE_LUTs_150827.tar.gz \
tle ..../testdata/input/drl.tle.2015082813 \
polar ..../testdata/input/off_USNO-PolarWander-UT1-ANC_Ser7_USNO_000f_20150828_201508280000Z_20150828000003Z_ee20150904120000Z_np.ascii \
leapsec..../testdata/input/leapsec.2015083101.dat \
compress off

```

The following script shows an example of a command line to run the ATMS SDR algorithm from the testscripts directory:

```

$ ..../wrapper/ATMS_C-SDR/run \
atms.rdr \
..../testdata/input/RATMS-RNSCA_npp_d20150828_t1948177_e1956177_b19869_c20150829015641940338_noaa_ops.h5 \
atms.fatms ..../testdata/output/FATMS.h5 \
atms.gatmo ..../testdata/output/GATMO.h5 \
atms.satms ..../testdata/output/SATMS.h5 \
atms.tatms ..../testdata/output/TATMS.h5 \
sdr.lut ..../testdata/input/ATMS-SDR_BE_LUTs_150416.tar.gz \
tle ..../testdata/input/drl.tle.2015082813 \
polar ..../testdata/input/off_USNO-PolarWander-UT1-ANC_Ser7_USNO_000f_20150828_201508280000Z_20150828000003Z_ee20150904120000Z_np.ascii \
leapsec..../testdata/input/leapsec.2015083101.dat \
compress off

```

The following script shows an example of a command line to run the CrIS SDR algorithm from the testscripts directory:

```

$ ..../wrapper/CRIS_C-SDR/run \
cris.rdr..../testdata/input/RCRIS-RNSCA_npp_d20150828_t1948177_e1956177_b19869_c20150829015638635404_noaa_ops.h5 \
cris.gcrso ..../testdata/output/GCRSO.h5 \
cris.rgcrs ..../testdata/output/RGCRS.h5 \
cris.rgtrs ..../testdata/output/RGTRS.h5 \
cris.scris ..../testdata/output/SCRIS.h5 \
sdr.lut ..../testdata/input/CRIS-SDR_BE_LUTs_150416.tar.gz \
tle ..../testdata/input/drl.tle.2015082813 \
polar ..../testdata/input/off_USNO-PolarWander-UT1-ANC_Ser7_USNO_000f_20150828_201508280000Z_20150828000003Z_ee20150904120000Z_np.ascii \
leapsec..../testdata/input/leapsec.2015083101.dat \
compress off

```

A successful execution usually requires a few minutes or more, depending on the speed of your computer and the size of the input. If execution fails, you will see an error message indicating the cause of failure (e.g., a file cannot be found, or a label cannot be recognized). Correct it and run again. If the problem has some other cause, it can be identified using the log files. Log files are automatically generated within the directory used for execution. They start with stdfile* and errfile* and can be deleted after execution. Other log and intermediate files may be generated automatically within the directory used for execution. They are useful for traceability and debugging purposes. However it is strongly recommended that users clean up log files and intermediate files left behind in the run directory before initiating a fresh execution of the SPA. Intermediate files from a previous run may affect a successive run and produce ambiguous results. The 'run' can be executed from any directory the user chooses. This can be done by prefixing it with the file path for the 'run' script.

NOTES:

1. The optional interface for cross-granule RDR file inputs for VIIRS_C-SDR currently requires that all RDRs used are single-granule RDRs only (viirs.rdr.prev, viirs.rdr, viirs.rdr.next).
2. In order for the cross-granule RDR file inputs to be used in processing, both ".prev" and ".next" RDR files must be provided. An "all-or-nothing" approach is used in which both ".prev" and ".next" must be provided, or neither of them are used during processing.
3. The input ATMS RDR must have a minimum of 3 granules for successful ATMS SDR generation.
4. The input CrIS RDR must have a minimum of 9 granules for successful CrIS SDR generation.
5. The TLE file must be within 14 days of the input RDR file. Use the TLE closest to, but prior, to the date of the input RDR file. The TLE files provided by the DRL are time-stamped as follows: drl.tle.yyyymmddhh.
6. The Polar Wander file must be within 30 days of the input RDR file. Use the Polar Wander file closest to, but prior, to the date of the input RDR file. The Polar Wander files provided by the DRL are time-stamped as follows: off_USNO-PolarWander-UT1-ANC_Ser7_USNO_000f_yyyymmdd*.ascii
7. If the Big-Endian (BE) LUT collection set tar file is not provided in the command line, the SPA uses the default SDR BE LUT collection set included with this release. Use the SDR BE LUT collection set that is closest to, but prior, to the date of the input RDR file. The LUT collection sets are time-stamped as follows: {Sensor}-SDR_BE_LUTs_yyyymmdd.tar.gz, where {Sensor} is 'VIIRS', 'CrIS', or 'ATMS'.
8. Leapsec ancillary files are cumulative. Use the latest leapsec file available regardless of the RDR date. The leapsec files provided by the DRL are time-stamped as follows: leapsec.yyyymmddhh.dat

9. The data products generated by this SPA may be visualized with the DRL's H2G_SPA (Hierarchical Data Format [HDF] to Georeferenced Tagged Image File Format [GeoTIFF] Converter Science Processing Algorithm). H2G is designed specifically for Direct Readout applications to create geolocated GeoTIFF images, jpeg browse images, and png browse images for parameter datasets in SNPP products and EOS products. H2G_SPA and its User's Guide are available for download from the DRL Web Portal. Please refer to Appendix A for information on enabling image production for this SPA in IPOPP.
10. This SPA makes network connections to itself using the Fully Qualified Domain Name (FQDN) of the machine where it is installed. This can cause problems, for example when running on a machine without an internet connection. If you have problems running this SPA, put your machine's FQDN on the 127.0.0.1 entry of its /etc/hosts file:
127.0.0.1 localhost.localdomain localhost mybox.myorg.org
where "mybox.myorg.org" is the added FQDN.

To Use the Scripts in the testscripts Directory

One simple way to run the algorithms from the directory of your choice using your own data is to copy the run-* .sh scripts from the testscripts directory to the selected directory. Change the values of the variables like WRAPPERHOME, INPUTHOME and OUTPUTHOME to reflect the file paths of the wrapper directories and the input/output file paths. Then modify the input/output file name variables. Run the script to process your data.

Appendix A

SPA Services

Installation of this SPA in IPOPP mode will make the SPA services listed in Table A-1 available to IPOPP. These services along with any other Prerequisite services (listed in Table A-2) will need to be enabled to allow IPOPP to automate production of the C-SDR_SPA data products. Furthermore, users who wish to generate image products from the data products generated by this SPA will need to enable the image-generating services listed in Table A-3. The SPAs containing the Prerequisite and the image-generating services listed in Tables A-2 and A-3 can be downloaded from the DRL Web Portal, in case they are not already available in your IPOPP installation. Details about these other SPAs are available in the respective SPA User's Guides. Please refer to the IPOPP User's Guide for instructions on how to install an SPA in IPOPP and enable the corresponding services.

Table A-1. SPA Services

Services for this SPA	Data Products Produced					
	Product Name	Destination (when installed in IPOPP)				
ATMS_C-SDR	ATMS SDR, TDR, and Geolocation Products	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">Product Name</td> <td style="padding: 2px;">Destination (when installed in IPOPP)</td> </tr> <tr> <td style="padding: 2px;">ATMS SDR, TDR, and Geolocation Products</td> <td style="padding: 2px; vertical-align: top;"> /raid/pub/gsfcdata/npp/viirs/level1/FATMS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5¹ /raid/pub/gsfcdata/npp/viirs/level1/GATMO_npp_dyymmdd_thhmmssS_ehhmmssS*.h5¹ /raid/pub/gsfcdata/npp/viirs/level1/SATMS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5¹ /raid/pub/gsfcdata/npp/viirs/level1/TATMS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5¹ </td> </tr> </table>	Product Name	Destination (when installed in IPOPP)	ATMS SDR, TDR, and Geolocation Products	/raid/pub/gsfcdata/npp/viirs/level1/FATMS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹ /raid/pub/gsfcdata/npp/viirs/level1/GATMO_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹ /raid/pub/gsfcdata/npp/viirs/level1/SATMS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹ /raid/pub/gsfcdata/npp/viirs/level1/TATMS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹
Product Name	Destination (when installed in IPOPP)					
ATMS SDR, TDR, and Geolocation Products	/raid/pub/gsfcdata/npp/viirs/level1/FATMS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹ /raid/pub/gsfcdata/npp/viirs/level1/GATMO_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹ /raid/pub/gsfcdata/npp/viirs/level1/SATMS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹ /raid/pub/gsfcdata/npp/viirs/level1/TATMS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹					
CRIS_C-SDR	CrIS SDR and Geolocation Products	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">Product Name</td> <td style="padding: 2px;">Destination (when installed in IPOPP)</td> </tr> <tr> <td style="padding: 2px;">CrIS SDR and Geolocation Products</td> <td style="padding: 2px; vertical-align: top;"> /raid/pub/gsfcdata/npp/cris/level1/GCRSO_npp_dyymmdd_thhmmssS_ehhmmssS*.h5¹ /raid/pub/gsfcdata/npp/cris/level1/RGCRS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5¹ /raid/pub/gsfcdata/npp/cris/level1/RGTRS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5¹ /raid/pub/gsfcdata/npp/cris/level1/SCRIS_npp_dyymmmdd_thhmmssS_ehhmmssS*.h5¹ </td> </tr> </table>	Product Name	Destination (when installed in IPOPP)	CrIS SDR and Geolocation Products	/raid/pub/gsfcdata/npp/cris/level1/GCRSO_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹ /raid/pub/gsfcdata/npp/cris/level1/RGCRS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹ /raid/pub/gsfcdata/npp/cris/level1/RGTRS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹ /raid/pub/gsfcdata/npp/cris/level1/SCRIS_npp_dyymmmdd_thhmmssS_ehhmmssS*.h5 ¹
Product Name	Destination (when installed in IPOPP)					
CrIS SDR and Geolocation Products	/raid/pub/gsfcdata/npp/cris/level1/GCRSO_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹ /raid/pub/gsfcdata/npp/cris/level1/RGCRS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹ /raid/pub/gsfcdata/npp/cris/level1/RGTRS_npp_dyymmdd_thhmmssS_ehhmmssS*.h5 ¹ /raid/pub/gsfcdata/npp/cris/level1/SCRIS_npp_dyymmmdd_thhmmssS_ehhmmssS*.h5 ¹					

VIIRS_C-SDR	Product Name	Destination (when installed in IPOPP)
	VIIRS SDR and Geolocation Products	/raid/pub/gsfcdtdata/npp/viirs/level1/GDNBO_npp_dyyyym <i>mdd_thmmssS_ehhmmssS*.h5</i> ¹ /raid/pub/gsfcdtdata/npp/viirs/level1/GIMGO_npp_dyyyym <i>mdd_thmmssS_ehhmmssS*.h5</i> ¹ /raid/pub/gsfcdtdata/npp/viirs/level1/GITCO_npp_dyyyym <i>mdd_thmmssS_ehhmmssS*.h5</i> ¹ /raid/pub/gsfcdtdata/npp/viirs/level1/GMODO_npp_dyyyym <i>mmdd_thmmssS_ehhmmssS*.h5</i> ¹ /raid/pub/gsfcdtdata/npp/viirs/level1/GMTCO_npp_dyyyym <i>mdd_thmmssS_ehhmmssS*.h5</i> ¹ /raid/pub/gsfcdtdata/npp/viirs/level1/ICDBG_npp_dyyyym <i>mdd_thmmssS_ehhmmssS*.h5</i> ¹ /raid/pub/gsfcdtdata/npp/viirs/level1/IVCDB_npp_dyyyym <i>mdd_thmmssS_ehhmmssS*.h5</i> ¹ /raid/pub/gsfcdtdata/npp/viirs/level1/IVOBC_npp_dyyyym <i>mdd_thmmssS_ehhmmssS*.h5</i> ¹ /raid/pub/gsfcdtdata/npp/viirs/level1/SVDNB_npp_dyyyym <i>mdd_thmmssS_ehhmmssS*.h5</i> ¹ /raid/pub/gsfcdtdata/npp/viirs/level1/SVI{01-05}_npp_dyyyym <i>dd_thmmssS_ehhmmssS*.h5</i> ¹ /raid/pub/gsfcdtdata/npp/viirs/level1/SVM{01-16}_npp_dyyyym <i>dd_thmmssS_ehhmmssS*.h5</i> ¹

¹ Where *yyyy, mm, dd, hh* represents the year, month and day of month for start of swath; the first *hh, mm, ss, S* represents the hour, minutes, seconds and 10th of a second for the start of swath and the second *hh, mm, ss, S* represents the end time of the swath.

Table A-2. Prerequisite Services

Prerequisite SPA services	SPA in which they are available
N/A	N/A

NOTE: The services VIIRS-SDR and VIIRS_C-SDR must never be run simultaneously.

Table A-3. Image-generating Services

Image-generating services	SPA in which they are available
vndnbright-geotiff	H2G_SPA
vndnbday-geotiff	H2G_SPA
vtoatcolor-geotiff	H2G_SPA
vm12h5-geotiff	H2G_SPA
ATMS-SDR-geotiff	H2G_SPA
CrlS-SDR-geotiff	H2G_SPA

NOTE: Please refer to the H2G_SPA User's Guide for more details about the image products, including their locations and filename patterns when they are generated in IPOPP.